CSCI-513: Autonomous Cyber-Physical Systems

Units: 4
Fall

Location: TBD

Instructor: Jyotirmoy (Jyo) V. Deshmukh
Office: SAL 340
Office Hours: 11am to 12pm on Tuesdays
Contact Info: Email: jyotirmoy.deshmukh@usc.edu,
Skype: jyotirmoy.deshmukh
[Response within 24 hours]

Course Description
Autonomous ground and aerial vehicles, complex robots, medical devices, and smart infrastructure systems are all examples of cyber-physical systems. Such systems are characterized by two main aspects: (1) physical components (e.g. electrical, electronic, mechanical, hydraulic mechanisms), and (2) software that is used to control the behavior of the physical components. Increasingly, such CPS applications are seeking to become autonomous with the help of artificial intelligence components trained using machine learning algorithms. In this course, we study different aspects of such *autonomous cyber-physical systems* (ACPS).

To build software for ACPS, we need to understand various aspects that span multiple engineering disciplines. From a computer science perspective, developing software for ACPS requires basic knowledge of the models of computation, software architecture for autonomous systems, AI and machine learning algorithms used for perception, planning and decision-making, and most crucially approaches to formally verify and test system safety. From the perspective of control theory and robotics, an ACPS software developer needs to be aware of basic control theory and algorithms, algorithms for path planning, state estimation and reasoning about uncertainty. The course focus will be on broad knowledge of ACPS software development rather than a deep dive into any specific topic. Deeper investigations of particular aspects of autonomy will be encourage through homeworks and projects. The course will position you to gain the skills required for industrial development of autonomous systems, and will also enable you to think about research problems in autonomy, particularly pertaining to safety.

Learning Objectives and Outcomes

1. Gain basic familiarity with modeling for CyberPhysical Systems.
2. Learn how to develop software for a CPS using a model-based development approach.
3. Learn how to write formal requirements for CPS models and perform model-based testing and verification.
4. Learn various ingredients for autonomy based on AI and control techniques such as path planning, reinforcement learning, basics of computer vision, basics of control and reasoning about uncertainty.
5. Learn basics of the software stack for autonomous systems.

Recommended Preparation:

a) Knowledge of mathematical logic at the level of CSCI 170 , and the knowledge of control theory at the level of EE 301.
b) Basic Knowledge of Matlab®/Simulink®
c) Basic Knowledge of Python

Course Notes:
Course Structure. In this course, most of the teaching will be accomplished through lectures. In addition, we will have the following elements:

1. **Homework assignments:** We will have 5 homework assignments through the semester. Three assignments will be written, and two will be programming assignments.
2. **Project:** The course will have a project, with structure outlined below

Project Structure. The purpose of the class project is for you to practice model-based development of a cyber-physical system application and adding some elements of autonomy to the system. Students will work in teams of two or three. The general expectation from the project is as follows: Each team will create a physical model for a CPS application. The team will then develop a controller using this model with the goal of providing some level of autonomy to the chosen CPS application. Teams will have the freedom to choose any control strategy, ranging from control theoretic methods to AI-based techniques such as reinforcement learning, imitation learning, etc. The expected behavior of the closed-loop model will be formally specified using an appropriate requirement formalism. The team will use verification or testing techniques to build confidence in the satisfaction of the requirements by the closed-loop model.

A few examples of CPS applications that the students can choose are provided below:

1. A UAV application, such as a quadrotor drone.
2. Autonomous vehicle subsystems such as lane changing, collaborative merging on a highway, stop-sign detection, collision avoidance, autonomous intersection management, or platooning.
3. A medical device system such as a pacemaker, an automatic insulin delivery system.
4. A ground robot for delivery, exploration, mapping or other applications.
5. A controllable biological process or a controllable biochemical reaction.

Project Deliverables:

a) Presentations:
   When: Students will perform two presentations during the course. In Week 7, each team will give a 10-minute presentation on the mid-term project progress. In the finals week, each team will give a 20-minute final presentation. There will be two 2-hour slots reserved for the final presentation during the finals week (or one 4-hour slot).
   Format: The mid-term presentation is expected to contain the following key elements: (a) review of related work (not exceeding 30% of the presentation), (b) proposed CPS application and associated problem, (c) timeline for expected results. The final project presentation is expected to contain the following elements: (a) summary of problem definition and solution, (b) key results and findings, (c) conclusions and related work.
b) **Report**: Students will submit a final report by a specified deadline in the finals week. The report will be expected to be single-column text, single-spaced pages in font size not exceeding 11pt. The report is expected to be at least 8 pages and at most 10 pages, excluding references.

Project Timeline:

Week 2	Team member identification
Week 4	Project Proposal Due
Week 7	Mid-Term Project Progress
Finals Week	Final Presentation & Report

Grading Breakdown for the Project (Total weight = 40%):

Proposal Document	5%
Mid-term Project Progress	10%
Final Report/Demo	20%
Final Presentation	5%

Technological Proficiency and Hardware/Software Required
Students are highly encouraged to be well-versed with Matlab® and Simulink®. Homework assignments will expect students to write Matlab® code and use Simulink® models. Knowledge of Python would be beneficial if students plan to do their projects using Python libraries such as Tensorflow or PyTorch.

Required Readings and Supplementary Materials
The course does not have a prescribed textbook. The following books are recommended for supplemental reading:

1) Principles of Cyber-Physical Systems by Rajeev Alur, MIT Press.
2) Introduction to Embedded Systems - A Cyber-Physical Systems Approach, by Lee & Seshia, Second Edition, MIT Press. http://leeseshia.org
3) Principles of Model Checking by Christel Baier and Joost-Pieter Katoen, MIT Press.
4) Cyber-Physical Systems by Raj Rajkumar, Dioniso de Niz, Mark Klein, Addison-Wesley.

Description and Assessment of Assignments
There will be 3 written assignments and two coding assignments (using Matlab®/Simulink®). Assignments will be assessed for the completeness and correctness of answers. Coding assignments will be graded using instructor-provided grading scripts. Partial credit will be assigned wherever applicable.

Grading Breakdown
Including the above detailed assignments, how will students be graded overall? Participation should be no more than 15%, unless justified for a higher amount. All must total 100%.

Category		Points	% of Grade	
Homeworks	HW1	100	10	55
	HW2 (coding)	100	10	
	HW3	100	10	
	HW4	100	10	
	HW5 (coding)	100	15	
Project		100		40
Participation		100		5
TOTAL				100

Grading Scale (Example)

Course final grades will be determined using the following scale

A 90-100
A- 85-89
B+ 80-84
B 75-79
B- 70-74
C+ 65-69
C 60-64
F 59 and below

Assignment Rubrics

1. Homework assignments will be graded for correctness of answers and provided explanation/proofs. Partial credit will be given wherever applicable.
2. Projects will be graded for the technical depth, novelty, repeatability of the experiment performed. Projects with a greater use of the concepts learned during the course will receive a higher grade.
3. The class will use Piazza for online discussions related to the concepts covered in the class. Students will be expected to ask and answer questions during in-class lectures and participate in discussions on Piazza. The participation grade will be assessed on both forms of student involvement in the course.

Assignment Submission Policy

Assignments are expected to be turned in to the instructor/TA by 11:59.59pm Pacific Time on the deadline. There will be a 5% penalty for every late day for 7 days. Assignments submitted 7 days after the deadline will be returned with a zero grade.

Grading Timeline

Graded assignments will be returned to students in a time period not exceeding 2 weeks from the submission of the assignment.

Course Schedule: A Weekly Breakdown

	Topics/Daily Activities	**Deliverable/ Due Dates**
Week 1	Course overview, Introduction to Models of Computation and Finite State Machines	
Week 2	Timed and Dynamical Systems, Linear Control Theory, Stability Analysis	HW1 posted, Team Member Identification for Projects
Week 3	Nonlinear Control, Hybrid Dynamical Systems, Controller Design	HW1 due
Week 4	Observation, State Estimation, Review of Simulink/Matlab	HW2 , Project Proposals due
Week 5	Introduction to Linear Temporal Logic, Büchi Automata, Signal Temporal Logic, Writing Specifications and Testing	
Week 6	Safety Verification: Model Checking, Reachability and Barrier Certificates	HW2 due, HW3 posted
Week 7	Mid-term Project Progress Presentations, Abstraction-based Control Design	MT project progress slides due
Week 8	Autonomous System Software Stack, Probabilistic Models	
Week 9	Classical and Deep Reinforcement Learning	HW3 due, HW4 up
Week 10	Probabilistic Model Checking, Planning	HW4 due, HW5 up
Week 11	Perception Systems, Tutorial on Convolutional Neural Networks	
Week 12	Navigaton and Localization, Filters revisited	
Week 13	Self-driving car algorithms, Synchronization and Consensus	HW5 due
Week 14	V2V and V2X technologies and algorithms, Thanksgiving break	
Week 15	Security and Privacy for ACPS, Medical CPS	

Finals Week	Final Project Presentations	Final Project Report Due

Statement on Academic Conduct and Support Systems

Academic Conduct:

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in SCampus in Part B, Section 11, "Behavior Violating University Standards" policy.usc.edu/scampus-part-b. Other forms of academic dishonesty are equally unacceptable. See additional information in SCampus and university policies on scientific misconduct, policy.usc.edu/scientific-misconduct.

Support Systems:

Counseling and Mental Health - (213) 740-9355 – 24/7 on call
studenthealth.usc.edu/counseling
Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention.

National Suicide Prevention Lifeline - 1 (800) 273-8255 – 24/7 on call
suicidepreventionlifeline.org
Free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week.

Relationship and Sexual Violence Prevention and Services (RSVP) - (213) 740-9355(WELL), press "0" after hours – 24/7 on call
studenthealth.usc.edu/sexual-assault
Free and confidential therapy services, workshops, and training for situations related to gender-based harm.

Office of Equity and Diversity (OED)- (213) 740-5086 | Title IX – (213) 821-8298
equity.usc.edu, titleix.usc.edu
Information about how to get help or help someone affected by harassment or discrimination, rights of protected classes, reporting options, and additional resources for students, faculty, staff, visitors, and applicants. The university prohibits discrimination or harassment based on the following *protected characteristics*: race, color, national origin, ancestry, religion, sex, gender, gender identity, gender expression, sexual orientation, age, physical disability, medical condition, mental disability, marital status, pregnancy, veteran status, genetic information, and any other characteristic which may be specified in applicable laws and governmental regulations. The university also prohibits sexual assault, non-consensual sexual contact, sexual misconduct, intimate partner violence, stalking, malicious dissuasion, retaliation, and violation of interim measures.

Reporting Incidents of Bias or Harassment - (213) 740-5086 or (213) 821-8298
usc-advocate.symplicity.com/care_report
Avenue to report incidents of bias, hate crimes, and microaggressions to the Office of Equity and Diversity |Title IX for appropriate investigation, supportive measures, and response.

The Office of Disability Services and Programs - (213) 740-0776
dsp.usc.edu
Support and accommodations for students with disabilities. Services include assistance in providing readers/notetakers/interpreters, special accommodations for test taking needs, assistance with architectural barriers, assistive technology, and support for individual needs.